Using Advanced Computer Vision Algorithms on Small Mobile Robots

G. Kogut[a], F. Birchmore[b], E. Biagtan Pacis[a], H.R. Everett[a],
[a]Space and Naval Warfare Systems Center, 53560 Hull Street, San Diego, CA 92152
[b]University of California, San Diego

ABSTRACT

The Technology Transfer project employs a spiral development process to enhance the functionality and autonomy of mobile robot systems in the Joint Robotics Program (JRP) Robotic Systems Pool by converging existing component technologies onto a transition platform for optimization. An example of this approach is the implementation of advanced computer vision algorithms on small mobile robots. We demonstrate the implementation and testing of the following two algorithms useful on mobile robots: 1) object classification using a boosted Cascade of classifiers trained with the *Adaboost* training algorithm, and 2) human presence detection from a moving platform. Object classification is performed with an *Adaboost* training system developed at the University of California, San Diego (UCSD) Computer Vision Lab. This classification algorithm has been used to successfully detect the license plates of automobiles in motion in real-time. While working towards a solution to increase the robustness of this system to perform generic object recognition, this paper demonstrates an extension to this application by detecting soda cans in a cluttered indoor environment. The human presence detection from a moving platform system uses a data fusion algorithm which combines results from a scanning laser and a thermal imager. The system is able to detect the presence of humans while both the humans and the robot are moving simultaneously. In both systems, the two aforementioned algorithms were implemented on embedded hardware and optimized for use in real-time. Test results are shown for a variety of environments.

KEYWORDS: robotics, computer vision, car/license plate detection, SIFT, human presence detection, robotic security, UGV, UAV, autonomous, teleoperated, collaborative behavior

1. BACKGROUND

The presence of remote-controlled robots on the battlefield continues to escalate, becoming an integral part of our military's arsenal for life threatening scenarios such as addressing the IED threats. However, there exists a definite tradeoff between the value added by the robot, in terms of how it contributes to the performance of the mission, and the loss of effectiveness associated with the operator control unit (OCU). Therefore, a need exists to enhance the functionality and autonomy of the current teleoperated systems in order to expand military use and eventually reach the vision for human-robot military teams. The Technology Transfer Project at SSC San Diego serves to rapidly expand military robotic capabilities through rapid enhancement of the technology readiness levels (TRLs) of behavior and interface methods from the research environment for transfer to acquisition programs. The approach is to pursue autonomous solutions that are common across the UxV domain, as well as advance human-robot interface tools to decrease the OCU's burden on the operator by increasing its usefulness for situational awareness. To date, the Technology Transfer project has produced phenomenal results, contributing mostly to the navigational needs to move from tele-operated platforms to autonomous systems. In FY-05, the focus has shifted to optimize more payload technologies to further build upon the navigational behaviors and augment the OCU for better representation of the

world both the robot and soldier are operating in. Computer vision is one key area the Technology Transfer project is actively pursuing, as it contributes to the advancement of both platform and interface capabilities

2. OBJECT DETECTION AND RECOGNITION

In this paper, we define "object detection" to be the process of identifying the existence and location of a particular type of object in an image. For example, a video stream may be scanned for the presence and locations of soda cans with no regard to the brand name of each can. Similarly, we define "object recognition" to be the process of taking detected objects and further determining more specific information about these objects, such as the brand of soda. A common approach to generic object detection has been to employ a large database of images containing all of the objects that may need to be detected. For instance, a previous incarnation of object detection for *ROBART III* (Figure 1) employed a color-correlation-matching algorithm whereby subsets of live images were compared to a database of stored images via thresholded correlation matching[1]. When detecting an object in a subset of an image, this sub-image was often compared with every image in the stored database to determine the best match. A database with a large number of objects allows the classification algorithm to detect a wider variety of targets. An immediate drawback to this approach is that the efficiency of detection decreases as the number of detectable objects grows within the scope of the algorithm's detection capabilities. If object detection is to be invariant to scale and orientation, the efficiency of the classification algorithm will decrease even more. Instead of working to increase the number of objects that can be detected, we have focused on increasing the efficiency and robustness of the initial detection of a single type of object to facilitate a conglomeration of modular detection and recognition mechanisms.

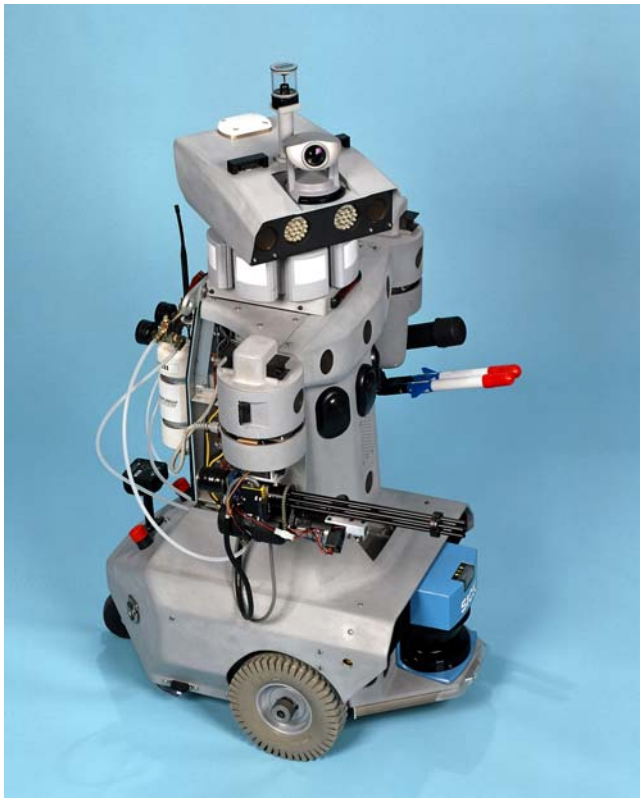

Figure 1: *ROBART III: The laboratory development platform for the Warfighter's Associate Concept[1], was initially developed to test the feasibility of automated response.*

When humans enter an area to search for some particular item, such as lost keys, we generally do not look at every possible location within that area and mentally compare it to every object we have previously seen. To determine if our lost keys are present, we might search for distinguishing features, such as the shiny key ring or the serrated edges of keys. Furthermore, we generally don't wander outside into our neighbor's yard looking for our keys if there is no reason to suspect they were lost there. When moving through areas where our lost keys are not likely to be, we more likely concentrate on detecting and recognizing objects that could cause us to trip and fall. We generally confine our searches

to likely places, such as a table in the kitchen, and do not consciously think about all of the other objects that we have previously observed.

Our approach to object detection for autonomous mobile robots shares many similarities with the aforementioned analogy of finding lost keys. Instead of comparing all sub-images of each image in question to all images in a database, we search each video frame for a group of known features for the target object by running each sub-image of the frame through a boosted cascade of weak classifiers. A cascade is an optimization method first proposed by Viola and Jones[2]. These classifiers are collectively trained using *Adaboost*, a method for improving the performance of any weak learning algorithm, developed by Freund and Schapire[11]. This approach is similar to the idea that when searching for lost keys, we often look for inherent features of keys that distinguish them from their surroundings. In addition, a logical alternative to comparing scenes from the area in question to all known objects is to use a small set of boosted classifiers to search for objects based on the context in which they can be expected to be found. This relates to the lost-keys analogy in that when searching from room to room, one would only look in probable locations, such as the kitchen table. This concept of context-based classification has been applied by Torralba, Murphy, and Freeman to only search for certain objects in relevant locations[3].

Our chosen method of object classification inherently possesses a large degree of modularity. If multiple objects are to be detected, then each cascade of weak classifiers could simply be swapped out with a new set that has been trained to detect each object in question. The detection efficiency of a trained set of boosted classifiers is sufficient to pass control to another more discriminating detection mechanism without compromising real-time performance. Some possible second-stage detection mechanisms could include using *ROBART III's* (Figure 1) bore-sighted laser to lock on and fire at a detected object[1], or utilizing a text-parser to interpret letters on a detected license plate or sign [10].

To test the robustness of *Adaboost*, we investigated the applicability of Dlagnekov's license-plate detection method to generic object detection[4]. To represent a generic object, we chose soda cans, making it possible to investigate the performance of our particular implementation of *Adaboost* with respect to scale, background clutter, and changes in specularity.

2.1 First experiment setup

2.1.1 Building the training set

Figure 2: *This is an example of an extracted frame where each soda can region has been manually labeled.*

We initially set up *Diet-Pepsi* cans around a room with a significant amount of clutter in the background. The cans were randomly placed at a variety of distances from *ROBART III*. Each can was rotated a different amount about its principal axis. This added rotation was used to extract features of the soda cans from a variety of angles and to reduce the influence of the prominent *Diet Pepsi* logo on the detection process. (If a robot's mission was to detect IEDs, the responsible parties would most likely not put their identifying logos on the outside of their products to provide convenient discriminating features.) Footage from *ROBART III*'s head-mounted camera was captured under three different lighting configurations (at a pixel resolution of 320x240 with a frame-rate of 3 frames-per-second) as three separate MPEG videos. Each video was then divided into two equal segments, and a portion from each segment pair was

randomly selected, resized to a resolution of 720x480, and extracted as a sequence of bitmap images. These images were set aside to build the training set. The remaining portions were merged together and set aside to be used as test footage for our final detection method. Each soda-can image was manually labeled in every 8th frame of the bitmap training images and sorted into four separate groups based on their image dimensions (Figure 2).

Instead of using a single strong classifier to detect soda cans at multiple scales, a separate strong classifier was trained for each scale. According to Dlagnekov's results, this approach yields a higher license-plate-detection rate when compared to a single strong classifier trained across multiple scales of training images[4]. Since soda-can images can exist in video footage at a multitude of scales with only a fixed window size for each classifier to detect them, using several different classifier scales helps reduce the amount of background clutter that may exist in situations where a smaller-scale soda can is present in a large input window. In order to further reduce the influence of background clutter on the detection process, negative training examples were used in conjunction with the positive training examples. Positive training examples are images of the object to be detected (i.e., soda-can images) and negative training examples are images of anything that is considered not to be a soda can (i.e., background clutter). In this experiment, negative training examples were randomly extracted from labeled images in areas that were not labeled as soda cans. An illustration of some positive training examples from an earlier experiment can be seen in Figure 3 below.

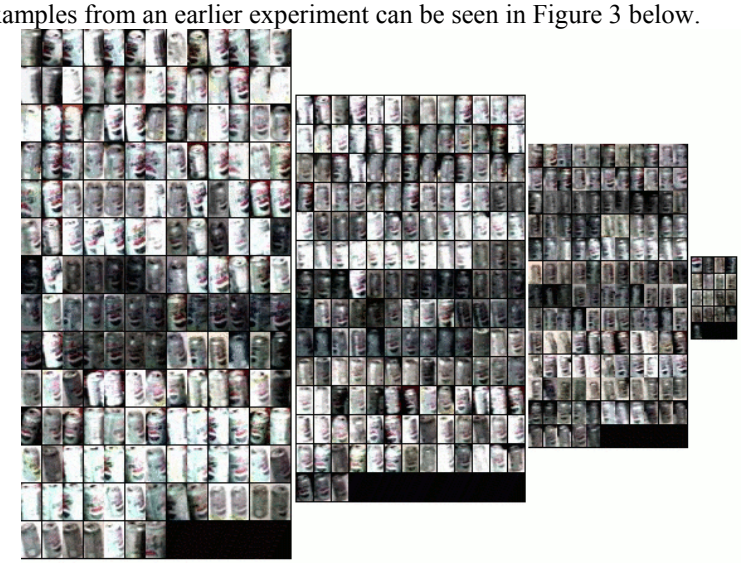

Figure 3: *This is a composite image of an earlier set of training images showing four different scales of acquired training images. Note that the black areas in the bottom-right regions of the composite images are blank, indicating no soda can images are present.*

2.1.2 Generating features

In order to train strong classifiers to detect soda-cans, a total of 2400 features were generated to create weak classifiers for *Adaboost*. These features were the same Haar-like features used by Dlagnekov in his license-plate detection algorithm[4]. Each feature consisted of dividing the input image into between two and seven equal vertical or horizontal portions. Each portion contains the mean of one of the following measurements of the image: X-derivative, Y-derivative, variance of these derivatives, or pixel intensities. The sum of the values in a particular set of the regions was then subtracted from the sum of the values in the remaining set of regions. This is similar to the way in which Chen and Yuille[10]detect text features in an image. Although we did not specifically extract text from the soda cans, these features did reasonably well in discriminating soda cans from background clutter during the *Adaboost* training phase.

2.1.3 Training strong classifiers

As mentioned before, a separate strong classifier was trained for each of four groups of soda-can training images. The sliding-window region for each scale of strong classifier is of size: 54x83, 30x47, 35x32, and 42x60. These window sizes were chosen by performing K-means clustering (for K=4) using the pixel width and height of each positive training image[14]. Each strong classifier was trained for 14 rounds to build a six-stage cascade of classifiers. Using a cascade greatly speeds up the object-detection process by rejecting a majority of non-soda-can regions. Instead of passing every

sub-region of the input image through all of the weak classifiers selected by *Adaboost* during the detection phase, most regions will be rejected after passing through the first few weak classifiers. In order to achieve a positive classification, an input region must pass through all stages of the cascade. However, this is a rare occurrence if we assume that most of the input image consists of non-soda can regions. A cascade of classifiers was also used by Dlagnekov to optimize his license-plate detection algorithm[4]. After training each strong classifier, the false-positive classifications from each were saved. Each strong classifier was then retrained using 1000 false positives from its previous training session. This approach allows each classifier to focus on training with "harder" examples, thereby increasing the classification rate of the resulting detection algorithm.

2.2. Second experiment setup

2.2.1 Building the training set

Figure 4: *This is an example of an extracted video frame that was used to build the training set.*

As a follow-up to the previous case, we performed an additional experiment to determine if Dlagnekov's license-plate detection method was effective enough to detect soda cans in real-time using two separate can orientations: horizontal and vertical. The training set was gathered by setting *ROBART III* at a fixed distance from a box (Figure 4). The background used was relatively static and contained a significant amount of clutter. Using this setup, 43 separate MPEG video files were captured, where each video was approximately 10 seconds long. Between each video capture, several *Diet Pepsi* cans were placed in a variety of locations around the box and rotated by various amounts, while maintaining either a horizontal or a vertical orientation. Each soda-can image was then hand-labeled to extract a total of 410 vertical and 420 horizontal images. Each of these labeled soda-can regions was next offset randomly by up to 1/8 of their width and 1/4 of their height to generate additional positive training examples, as Dlagnekov did for his license-plate detection method[4]. This resulted in 4,444 positive training examples of vertically oriented and 4,554 of horizontally oriented soda cans. In addition, each strong classifier was given 23,200 negative training examples that had been randomly extracted from video footage containing only the aforementioned box, with no soda cans present.

2.2.2 Training the classifiers

In contrast to the previous experiment, only two separate strong classifiers were trained: one for horizontally-oriented and one for vertically-oriented soda cans. The window sizes for each strong classifier were determined by taking the average window size of the positive training set for each can orientation. This resulted in window sizes of 17x27 pixels for the vertically-oriented soda cans and 30x14 pixels for the horizontally-oriented soda cans. Each classifier was then trained in the same manner as in the previous experiment, where 14 rounds of training were executed to build a six-stage cascade of classifiers. To minimize the effect of camera-lens distortion and to increase the efficiency of detection, only a central 161x184 sub-window of each image was used to collect negative training examples. This same sub-window was later used to define a sub-area in which to scan for soda cans during the detection process.

2.3 Results

2.3.1. Training error rates – experiment 1

After the four strong classifiers were trained, the results indicated that training was moderately successful with error rates indicated by the graph in Figure 5 below. However, since the set of strong classifiers must be sequentially applied to the input image, the cumulative false-positive rate of the strong classifiers has to be taken into account when determining the overall training-error rate, which results in a collective false positive training-error rate of 85.49 percent. In addition, each strong-classifier error rate merely indicates how well *Adaboost* was able to fit the weak classifiers to the training data, showing that the training data can be learned reasonably well using the current set of features. However, this gives little indication of the actual performance of the strong classifiers on new input images. The global error rate for the first layer of each cascade is shown in Figure 5. In this graph, the dark black line corresponding to the strong classifier with a 54x83 window consistently has the lowest training error. This makes sense, because a larger soda-can image contains more pixel information than a smaller image, which implies that larger images would have more discernable features than smaller images. In fact, the very first weak classifier chosen has a weighted training error rate of only 15 percent, while the first weak classifiers corresponding to detection windows of size 35x32 and 30x47 have error rates of roughly twice this amount. Most of the error exhibited in Figure 5 is a result of false-positive classifications. The false-negative rates for all four window scales start at less than 1 percent and drop very quickly towards zero after only a few rounds of training. However, the false-positive rates for all four window scales start at between 15 percent and 35 percent and decline to between 1 percent and 7 percent. This seems to indicate that the features we chose to train *Adaboost* are not extracting enough information from the lower-resolution soda-can images, causing the weak classifiers to have a somewhat difficult time discriminating between soda cans and their background. For these four scales of strong classifiers, the first stage of each cascade was trained for 14 rounds. *Adaboost* could be trained for more rounds to decrease this error rate. The main drawback to doing this is that each region being classified would have to be run through more classifiers, which would significantly decrease the performance of this detection method.

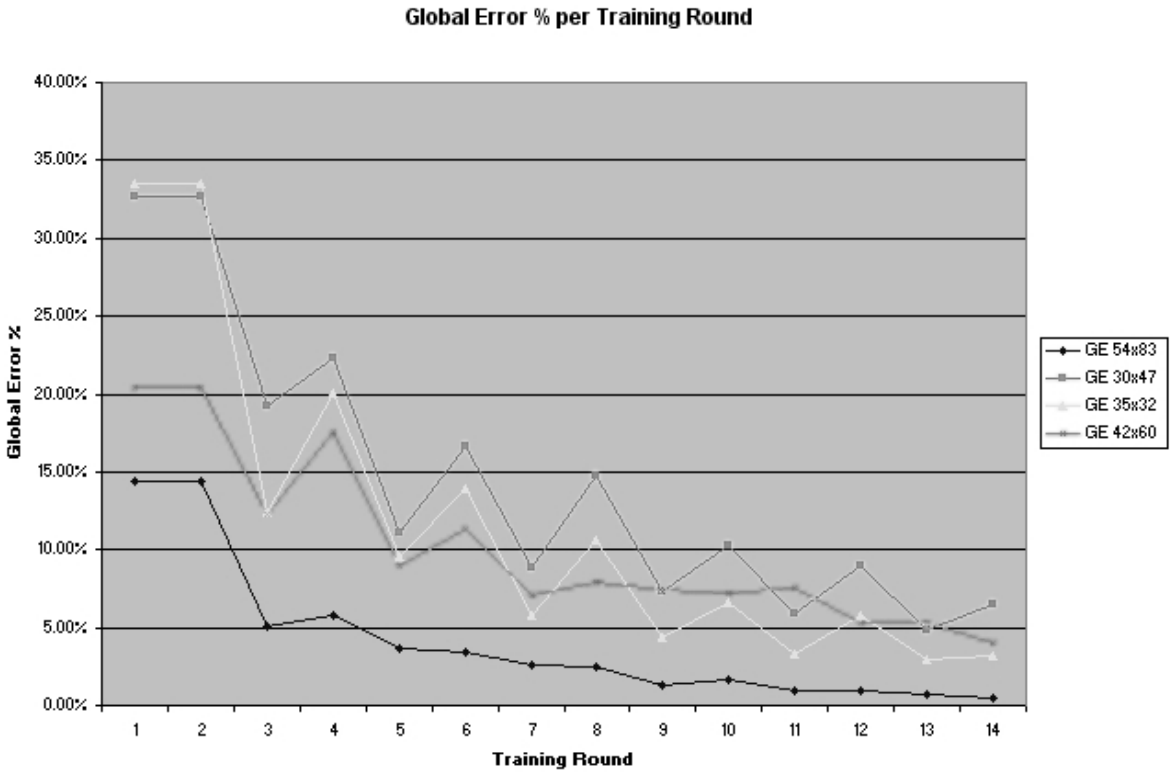

Figure 5 *Global error rates from training four different strong classifiers for soda cans.*

2.3.2 Best features selected by *Adaboost* – experiment 1

Among the top four features selected by *Adaboost* (Figure 6), none corresponded purely to pixel brightness values. Dlagnekov's experience was similar, which he attributed to the variation in lighting conditions[4]. The first feature selected by *Adaboost* takes the average X-derivative from the black regions and subtracts it from the average derivative of the blue regions. Intuitively, this might correspond to the vertical edges of the can or the region containing its logo. The second feature selected by *Adaboost* takes the mean of the difference in the variance of the Y-derivative from the image region to be classified. This particular feature appears to take into account the top and bottom of the soda can to perform classification.

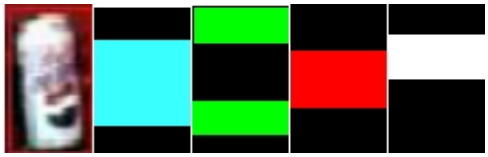

Figure 6 A 58x83 soda-can image followed by the top four weak features selected by *Adaboost* for the classifier of that scale. The colors represent: Blue - mean of X derivative, Green - mean of variance of Y derivative, Red - mean of Y derivative, White – mean of variance of X derivative, Black - negative of colored regions.

2.3.3 Test results – experiment 1

In order to test the performance of the four strong classifiers, we ran each separately on the same set of video footage, which consisted of images of soda cans that were captured during the training stage, but not labeled for use in the training set. The scenario used for testing involved *ROBART III* moving towards another robot (the iRobot *ATRV* shown in Figure 7) with soda cans placed on and around it. *ROBART III* then backed away and headed towards a *Pepsi* vending machine with soda cans sitting on the floor in front of it. We ran each strong classifier on the test video using Dlagnekov's detection framework, and adjusted the threshold value on the strong classifier until a reasonable balance was achieved between true-positive and false-positive classifications. Each positively classified region in each frame of the test video was then marked with a yellow rectangle. By visually inspecting each frame of the marked video footage for the frequency of correctly placed yellow rectangles, the frames containing the highest ratio of true-positive to false-positive classifications were identified. Two of these frames are shown below in Figures 7 and 8.

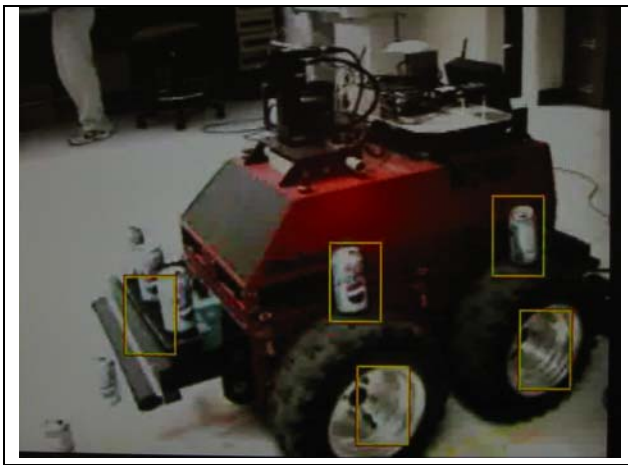

Figure 7 54x83 with threshold 1.3

Figure 8 30x47 with threshold 1.5

The strong classifier at scale 54x83 (Figure 8) with a threshold value of 1.5 seemed to perform the best overall for classifying soda cans of a similar scale. During training, this strong classifier similarly performed better than the others at learning its training set. Once again, this is probably due to the fact that this classifier was run on images that were larger than the others and thus contained more discriminate features. Another reason for the high classification rate of this strong classifier could be the use of clustering applied to each detected location. To reduce the number of false positives, all regions that have been classified as containing soda cans are clustered. If more than five classifications exist in a particular cluster, then the average location in this cluster is marked with a yellow rectangle. This technique decreases the number of false positives that may result from noise. One drawback to this method is that true-positive classifications

may be off-center as a result of false-positive classifications biasing each cluster centroid, which may explain the left-most rectangle in Figure 7. This clustering method was also used by Dlagnekov in his license-plate detection method[4].One area of the background that consistently caused false-positive classifications was the metal wheels of the *ATRV* robot, as seen in Figure 7. This may be attributed to the similarity between the specularity of the soda cans and the metal hubs. The image in Figure 8 exhibits the best classification rate achieved in this experiment. This high classification rate is most likely due to the relative sparseness of the background, making our chosen set of features suitable for distinguishing soda cans in this particular image.

2.3.4 Training error rates – experiment 2

After the two strong classifiers were trained, the training error rates were fairly low as shown in Figure 9 below. When compared to the results from the previous experiment, the global error rates of these 17x27 and 30x14 strong classifiers are approximately the same as those of the 35x32 and 42x60 strong classifiers. However, the 54x83 strong classifier from the previous experiment had a significantly lower training error rate than either of the strong classifiers in this experiment. This is most likely due to the aforementioned fact that higher resolution images contain more usable information than lower resolution images, causing higher resolution images to have more distinct features.

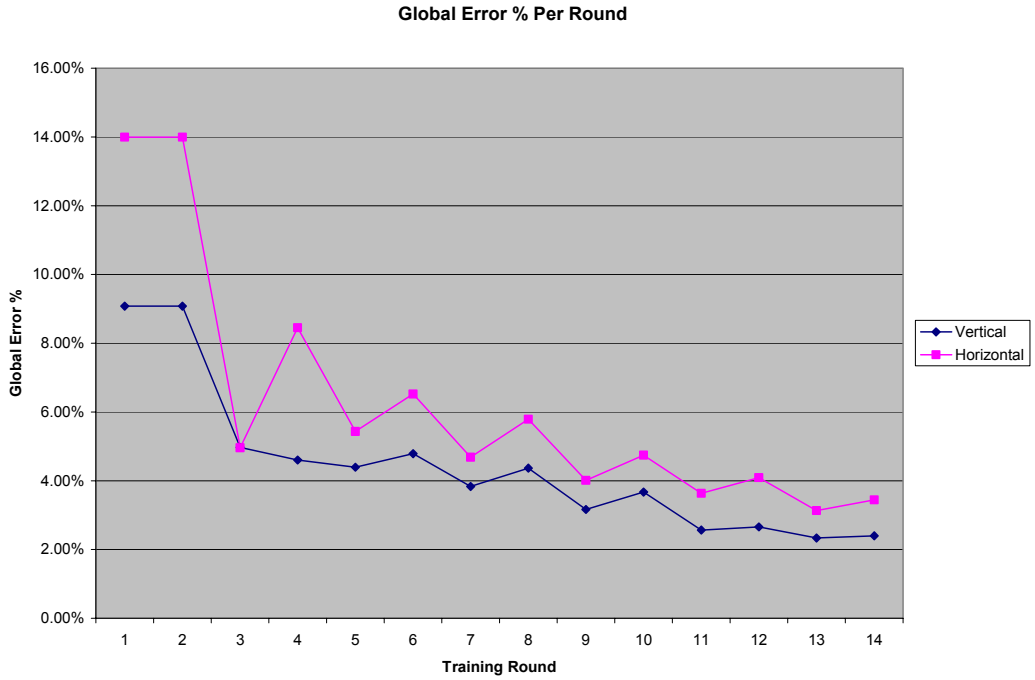

Figure 9 Global error rates from training two different strong classifiers for soda cans. The window size of each classifier is 17x27 and 30x14 for vertically and horizontally oriented soda cans, respectively.

2.3.5 Test results – experiment 2

In order to test the performance of the two strong classifiers, each strong classifier was executed on video footage that had been excluded from the training set prior to training. The strong classifiers were executed sequentially on the test video using a modified version of Dlagnekov's license-plate detection framework. As in the previous experiment, each location classified as containing a soda can was stored in a set of 20 clusters. If any particular cluster exceeded a threshold of 50 detections, then the centroid of that cluster was determined to contain a soda can. A yellow rectangle was then drawn at each average cluster location in each image being classified. One drawback to this method of clustering is that each strong classifier can only detect a maximum of 20 individual soda cans. One possible solution to this problem would be to apply K-means clustering[14] to the regions classified as containing soda cans, though this would decrease the

efficiency of our detection method. Using this setup, the central region of each image in a stream of video was scanned and marked at locations determined to contain soda cans. In one particular set of test footage, a 100-percent detection rate was achieved in real time with no false positives (Figure 10).

Figure 10: Real-time detections of test video footage from *ROBART III*'s camera

Surprisingly, the detection rectangles around the horizontally-oriented soda cans are lined up better than the rectangles around the vertically-oriented soda cans. This is most likely a result of lens distortion. One obvious manifestation of this lens distortion can be seen by observing the shape of the wooden box in the center of the image. In reality, the box has flat parallel edges, yet it appears convex in this image. The offset rectangle around the vertically-oriented soda can at the top of the image probably resulted from averaging a cluster consisting of true-positive and false-positive detection regions. The offset rectangle at the bottom of the image was initially aligned very well with the soda can. In the later portion of the video from which this frame was extracted, this rectangle drifts to the upper-right. This is probably a result of a combination of noise in the video stream with the perspective distortion of this particular soda can. Despite these anomalies, the detection windows are still aligned well enough to extract coordinates that allow *ROBART III* to aim the pneumatically-powered weapon surrogate.

3. MOTION AND HUMAN-PRESENCE DETECTION

The need to detect motion from a moving platform has become an important requirement for robotics. As robots become increasingly common in populated areas, their ability to effectively co-exist with humans will depend on their ability to perceive and process information about surrounding activities. For example, physical security robots have the specific mission of detecting and reporting motion.

While motion detection for physical security applications involving static cameras is now a common and mature technology, it is not a solved problem for mobile applications. The problem of detecting motion from a moving platform is substantially more complex since a mobile robot must consider its constantly changing visual surroundings and be capable of detecting motion while it is on the move. Robust solutions to this challenging problem may require detailed knowledge of robot motion and a model of the surrounding environment. A variety of vision-based techniques have been applied to motion detection on the move with some success[6, 7]. However, most of these techniques are either too computationally complex for integration on small, i.e. man-portable, robots or make assumptions about the environment that limit their use.

3.1 Laser/vision data fusion

We approached the problem of motion detection on-the-move by focusing on detecting human presence from a moving robot in cluttered indoor and outdoor environments. This project was sponsored by the Defense Threat Reduction Agency in collaboration with the University of Texas Applied Research Laboratory, Austin, TX, and the Idaho National Laboratory, Idaho Fall, ID. To accomplish the task of detecting human presence from a moving platform with limited computational resources, we used two-stage sensor fusion. The first stage involves using a scanning laser to detect initial changes, while second stage involves applying image processing to thermal imaging data to verify any potential human presence.

The Idaho National Laboratory (INL), SSC SD's strategic partners under the Technology Transfer project, adapted ladar-based simultaneous localization and mapping (SLAM) technology developed by the Stanford Research Institute, yielding a change-detection-on-the-move capability referred to as *real-time occupancy change analysis* (ROCA) [12]. Once a map representing the operating area is produced, the robot uses the occupancy grid from the SLAM algorithm to detect subsequent changes. Because change detection is based on an adaptive and persistent background model, the disparities that occur outside the sensor range of the robot are detected when the robot re-visits the area. The location of each perceived disparity can be sent as a vector to supporting sensors for further assessment.

The ROCA algorithm performs an excellent job of detecting physical changes in the environment. Ad-hoc testing results indicate a very good detection rate with a low false-alarm rate for detecting changes on the scale of a human moving within approximately 20 meters of the robot. However, change detection alone is not sufficient for human presence detection, as moving changes in the environment, such as re-arranged furniture or waving tree limbs, are flagged as changes.

Therefore, a simple data-fusion method known as time-gated windowing was explored for use in determining if the observed change in the environment was due to human presence. This method is based on two assumptions: 1) people will emit more thermal radiation than their surroundings and 2) the aspect ratio of height to width of the thermal "hot spot" emitted by a human is 4:1 or greater. The time window is initiated by an occupancy change detected by the ROCA system. The robot's pan-tilt mechanism is used to direct a thermal imager at the detected change, and several snapshots are recorded as digital images. The acquired images are then segmented by extracting "hot spots" from each image using an empirically calculated threshold value. (This threshold value was determined offline by analyzing thermal images of humans at various distances from the thermal imager.) The vertical aspect ratio of each "hot spot" is then calculated, and ratios of 4:1 or greater are designated as likely human presence.

3.2 Thermal verifier experimental setup

A simple proof-of-concept experiment was performed to verify the viability of using ROCA and a thermal imager as a robot payload for human presence detection. The experiment consisted of having people walk in front of the test robot, and verifying three properties of the system: 1) that human movement was correctly detected as occupancy changes by ROCA, 2) that the thermal imager could be aimed accurately enough at a moving person to acquire a quality thermal image, and 3) that the thermal human-verification algorithm confirmed that a human was present in the snapshot images. The experiment took place indoors, and involved five human test subjects who each performed three trials. First the robot was allowed to explore an open lab area with an area of approximately 20meters by 25meters, and develop a map representation of the lab. Each subject was then introduced into the lab area, one at a time, and instructed to move however they wanted and then leave the lab area. The output of the data fusion system was logged for analysis.

3.3 Thermal verifier results

Five people each performed three trials, for a total of 15 trials. The SLAM-based change detection system had a 0% false alarm rate and 100% detection rate during the limited experiment. That is, all changes indicated by the robot were actual changes to the environment (moving humans). The thermal human presence verification algorithm reported 5 false alarms while left running continuously during the experiment. However, because of the accuracy of the ROCA system, the overall system had a 0% false alarm rate. For the overall system to report a false alarm, both sensor components must simultaneously report false alarms. It should be noted, however, that these results were taken from a very limited experiment conducted in ideal lab conditions. Further, more extensive testing in an uncontrolled, outdoor environment is required to verify the effectiveness of the system in a real-world application.

4. TELEOPERATED UAV CONTROL

Computer vision algorithms allow the development of new capabilities for robotic systems. We demonstrate ROBART III teleoperating a remote-controlled blimp, shown in figure 8. The remote control radio link is interfaced to ROBART III's digital IO control, allowing ROBART III to teleoperate the UAV with adjustments to velocity and altitude. ROBART III tracks the position of the blimp through an object detection and classification system similar to that described in 2. Our initial efforts focused only on having ROBART III maintain the blimp's altitude. Further development will allow ROBART to teleoperate the blimp to given waypoint positions.

Figure 11 ROBART III teleoperating a 3-foot diameter Mylar aerostat

5. CONCLUSIONS

5.1 Object detection

We have determined that Dlagnekov's license-plate detection method is fairly robust for detecting soda cans without additional modifications to the set of generated features. We have, however, identified several shortcomings that suggest further research is required to pursue our long-term goal of achieving generic real-time object detection and recognition for mobile robots. Our results so far seem to indicate that an *Adaboost*-based implementation using multiple strong classifiers performs well with real-time efficiency against a somewhat static background. However, backgrounds against which the strong classifiers have not been trained cause numerous false-positive classifications to occur. In addition, the amount of time required to detect multiple classes of objects in each input image increases very rapidly as additional strong classifiers are sequentially employed, thereby undermining the real-time efficiency of our detection method. Furthermore, the introduction of multiple strong classifiers would reduce the robot's ability to eliminate false-positives, since the overall false-positive rate of the detection method is the sum of the false-positive rates from successively applied strong classifiers. Despite these shortcomings, a long-term solution to the problem of achieving real-time generic object detection and recognition for mobile robots may in part be solved through the use of an *Adaboost*-based object-

detection method. Although our detection method would not work well as the primary method for detecting objects in a cluttered environment, it might work well as an intermediate layer of detection.

5.1 Motion and human-presence detection

Sensor fusion has been a key method of achieving reliable performance in robotics, a common example being the Kalman filter used to fuse data from multiple navigation sensors and improve the accuracy of robot navigation and localization[13]. However, sensor fusion has been less rigorously applied to computer vision applications. Computer vision applications for robots typically suffer from two shortcomings: computational complexity and the failure to gracefully handle changes in perspective, scale, and lighting experienced in dynamic, unpredictable real-world environments. Thermal verification demonstrates how even simple data fusion can assist in addressing these shortcomings. The primary driver of the computational complexity of computer vision applications is the large visual area processed. Sensor fusion can reduce image space dramatically, allowing vision routines to focus on small area rather than an entire scene. In this case, the scanning laser reduces the search space of the thermal verifier in both time and space. The computer vision routine only runs when motion is detected by the laser, and operates over a much smaller area, defined by the laser detected disparity. This reduces the computational complexity by at least an order of magnitude as compared to a vision system continuously searching the entire scene continuously. This approach is analogous to some human behaviors, in that we only visually inspect things closely when cued by unusual noises or motions.

6. FUTURE WORK

In this initial effort on generic object detection, we focused mainly on applying simple and easy to compute features to every sub-window of an input image from a single camera. Like humans, robots often use multiple sensors to interact with their environment. Some of these sensors may be used to offset a majority of the workload from an *Adaboost*-based implementation. This would enable an *Adaboost*-based implementation to use slower-to-compute yet more generalized features such as the opponent-color features used by Jain and Healey[5]. For example, additional sensors could be used to detect an anomaly while an *Adaboost*-based algorithm performs detection only on the region that is classified as an anomaly, as will be discussed in section 3.

Dlagnekov's license-plate detection method performs well in terms of efficiency and robustness to background clutter under limited conditions. However, this detection method seems to come up short in terms of rotation-invariance, detection rates against multiple backgrounds, and efficiently detecting numerous classes of objects. To improve invariance to rotation and scale, a method such as D. Lowe's Scale-Invariant Feature Transform (SIFT) [9] or Sivic and Zisserman's *Video Google*[8] could be investigated. As Dlagnekov observed when detecting the make and model of automobiles, SIFT can be used to achieve very high detection rates at the expense of a hefty performance cost when detecting a large variety of objects. For instance, Dlagnekov ran SIFT using a database of 1,102 different automobile images and achieved a classification rate of 89.5 percent, where each object classification took 30 seconds[4]. A SIFT-based implementation may be able to achieve real-time detection rates if the number of objects in its database is reduced while other supplementary methods are used to maintain overall robustness.

For instance, some sort of context-based classification[3] could be used to limit the set of classifiable objects based on the environment in which they can be expected to be found. In addition, the locations of an image in which to search for objects of interest could be narrowed down by utilizing data collected from other sensors on the robot. Numerous *Adaboost* classifiers with lower detection rates could also be used to significantly reduce the number of regions of an image which should be searched for objects. These classifiers could be trained using a very low number of detection layers that still eliminate most of the background regions. Regions detected by each classifier could then be passed to a more accurate detection algorithm or its location could be acquired and passed to another sensor on the robot for further processing. While *Adaboost* alone may not be suitable for generic object detection, it may at least play a supporting role in the future of generic object recognition for mobile robots.

From our research, we have developed a soda-can detector that works well under somewhat ideal circumstances. Despite these constraints, the results show promise for detecting some objects well in certain contexts. Our detection method

could serve as an intermediate layer in the overall detection and classification of signs on doors. For instance, ROBART III could first use its laser scanner to detect the presence of a door. It could then precisely align itself with the door to provide conditions conducive to detecting objects using the boosted classifier mentioned in this paper. This classifier could then be used to detect signs on the door, which is a simpler problem than detecting soda cans. Once a sign is detected, the words on the sign could be interpreted using an optical character recognition algorithm. This way, conditions that facilitate optimal performance for our detection method could be optimally provided by the robot. Rather than focusing on a large catch-all object-detection method, we have concluded that a robot may be better able to interact with its environment if it is equipped with many smaller, specialized detection methods that can be applied appropriately once the robot establishes a suitable context for the deployment of each. Our future work will focus on performing robotic vision using the full capabilities of the robot's navigational and sensing capabilities, rather than performing computer vision as an isolated task.

7. REFERENCES

1. H.R. Everett, E.B. Pacis, G. Kogut, N. Farrington, S. Khurana. Towards a Warfighter's Associate: Eliminating the Operator Control Unit. Space and Naval Warfare Systems Center, San Diego (SSC San Diego) University of Southern California (USC)
2. P. Viola, M. Jones. Rapid object detection using a boosted cascade of simple features. Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on , Volume: 1, 8–14 Dec. 2001 Pages:I-511 - I-518 vol.1
3. A. Torralba, K. Murphy, W. Freeman, and M. Rubin. Context-based vision system for place and object recognition. In Intl. Conf. Computer Vision, 2003.
4. Louka Dlagnekov. Video-based Car Surveillance: License Plate, Make, and Model Recognition. Masters Thesis, University of California, San Diego, 2005.
5. A. Jain, G. Healy. A Multiscale Representation Including Opponent Color Features for Texture Recognition. In IEEE Transactions on Image Processing, Vol.7, No.1, January 1998
6. N. Papanikolopoulos, P. Khosla, T. Kanade, ``Visual tracking of a moving target by a camera mounted on a robot: a combination of control and vision", IEEE Trans. Robotics and Automation, Vol 9, No 1, February 1993
7. Randal C. Nelson, ``Qualitative Detection of Motion by a Moving Observer", Proc. IEEE Conference on Computer Vision and Pattern Recognition, Maui Hawaii, June 1991, 173-178.
8. J. Sivic and A. Zisserman. Video google: a text retrieval approach to object matching in videos. Proc, ICCV, 2003.
9. D.Lowe. Distinctive image features from scale-invariant key-points. LJCV, 2(60):91–110, 2004.
10. A. Chen, X. Yuille. Detecting and reading text in natural scenes. *CVPR*, 2:366–373, 2004.
11. Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. In *European Conference on Computational Learning Theory*, pages 23–37, 1995.
12. Kurt Konolige, Ken Chou: Markov Localization using Correlation, IJCAI 1999: 1154-1159
13. Bruch, M.H., Gilbreath, G.A., Muelhauser, J.W.,and J.Q. Lum, "Accurate Waypoint Navigation Using Non-differential GPS," AUVSI Unmanned Systems 2002, Lake Buena Vista, FL, July 9-11, 2002.
14. J. B. McQueen (1967): "Some Methods for classification and Analysis of Multivariate Observations", Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, University of California Press, 1:281-297

SPIE Proc. 6230: Unmanned Systems Technology VIII,
Defense Security Symposium, Orlando, FL, April 18-20, 2006